## IN THE CLAIMS

Please amend the claims as follows:

1. (currently amended) An assembler for processing structured assembly language expressions utilized in structured assembly language programming, said assembler comprising:

program code means for recognizing a structured assembly language expression's mnemonics containing elements arg1 cc arg2, wherein said cc is a condition code, wherein the form of said expression's mnemonics or the nature of one or more of said expression's elements selects a corresponding comparison opcode, wherein said arg1 and said arg2 are valid arguments for said selected comparison opcode;

program code means for constructing a data structure referencing said arg1, said arg2, said cc, and a branch destination;

program code means for generating a comparison opcode in response to elements of said data structure;

program code means for generating a conditional branch based on said condition code in said data structure;

program code means for generating a first branch location for execution to proceed at as if said structured assembly language expression is true; and

program code means for generating a second branch location for execution to proceed at as if said structured assembly language expression is false; and

program code means for generating a third branch location for execution to proceed at to the end of said structured assembly language expression; and

program code means for indicating said branch destination in said data structure is a branch to said first, said second, or said third branch locations.

- 2. (original) The assembler of Claim 1, wherein said assembler further includes program code means for recognizing a structured assembly language expression's mnemonics having a form cc, wherein said cc is a condition code.
- 3. (original) The assembler of Claim 1, wherein said assembler further includes a program code means for generating a data structure referencing at least no arguments, cc, and a branch destination in response to said condition code.
- 4. (original) The assembler of Claim 1, wherein said assembler further includes program code means for not generating a comparison opcode in response to said data structure.
- 5. (original) The assembler of Claim 1, wherein said assembler further includes a program code means for generating assembly language code by iterating over a vector of said structured assembly language data structures of various forms.
- 6. (original) The assembler of Claim 1, wherein said assembler further includes

program code means for recognizing a structured assembly language expression's mnemonics resulting from a logical ANDing of SA\_Expr1 and SA\_Expr2, wherein each of said SA\_Expr1 and said SA\_Expr2 is a unit or a compound structured assembly language expression;

program code means for setting said branch in each data structure of said SA\_Expr1 that is branching to said first branch location to branch to end of said SA\_Expr1; and

program code means for concatenating and preserving order of data structures in said SA\_Expr1 and said SA\_Expr2 into a single compound structured assembly language expression.

7. (currently amended) The assembler of Claim 1, wherein said assembler further includes

program code means for recognizing a structured assembly language expression's mnemonics requiring a logical ORing of SA\_Expr3 and SA\_Expr4, wherein each of said SA\_Expr3 and said SA\_Expr4 is a unit or a compound structured assembly language expression;

program code means for changing said a branch location in each data structures of said SA\_Expr3's data structures, except for a last data structure of said SA\_Expr3's last data structure, from said second branch location to end of said SA\_Expr3;

program code means for complementing said branch condition in said SA\_Expr3's last data structure;

program code means for changing said branch location in said <u>last data structure</u> of said SA\_Expr3's last data structure from a branch to said first location to branch to said second location, or from a branch to said second location to branch to said first location; and

program code means for concatenating and preserving order of data structures in said SA\_Expr3 and said SA\_Expr4 into a single compound structured assembly language expression.

8. (currently amended) The assembler of Claim 1, wherein said assembler further includes

program code means for recognizing said structured assembly language expression's mnemonics requiring from a logical negation of SA\_Expr5, wherein said SA Expr5 is a unit or compound structured assembly language expression;

program code means for changing said a branch location in each data structures of said SA\_Expr5's data structures, except for a last data structure of said SA\_Expr5's last data structure, from said first branch location to said second branch location, while changing said branch location in each of said SA\_Expr5's data structures, except for said SA\_Expr5's last data structure, from said second branch location to said first branch location; and

program code means for complementing said branch condition in said SA\_Expr5's last data structure.